数据需求文档

古诗人时空流动性分析系统

# 系统简介

古诗人时空流动性分析系统，受到“唐宋文学编年地图”的启发，旨在通过利用古诗人的人生轨迹，结合现代轨迹嵌入技术进行分析与可视化展示。不同于“唐宋文学编年地图”，只将侧重点集中在了诗人人生轨迹的可视化与地理位置诗词的统计，本系统更侧重于通过轨迹数据进行数据仓库的建立和数据挖掘与数据分析，是结合前沿科技技术的对古诗人时空流动性模式进行挖掘和分析的系统。

本系统旨在提供一种通用的计算方法与可视化平台，基于此，任何领域的人群人生轨迹数据都可以通过此套流程进行分析和可视化。

本系统以中南民族大学王兆鹏的精标注唐宋数据为例，用作系统的主要案例进行分析和可视化展示。

# 功能对应的数据需求

本章节阐述因功能界面产生的数据要求。

## 界面布局及框图

需准备各个中/英文的翻译数据（即文案部分）。

* 含各个界面中（包含介绍、按键等部分）的文案。

## 原始数据

来源于唐宋编年史地图的原始数据，根据API所爬取到的数据，按格式进行说明。

### 诗人人生轨迹数据

按数据返回格式的层级进行说明：

* 地图可视化时的中心点位置坐标
* 地图可视化时的地图缩放级别
* IsMarked: 功能意义，无数据意义
* Title: 诗人以及其人生年份，数据处理时需拆分
* Detail: 引用书目，例如XX岁在XX地点进行了游历。以长字符串进行存储，实际处理时应当拆分，并使用array存储。
* Lines: Array with the objects of key “Markers” Array (Routes Points)
  + 对每个Routes Point
    - Id&RegionId
    - 坐标
    - 相对于左上的offset（无数据意义）
    - 可见性（无数据意义）
    - Titile: 地名
* Markers: Array with the objects
  + For each Location Point object
    - RegionID
    - 坐标
    - 相对于左上的offset（无数据意义）
    - 可见性（无数据意义）
    - Titile: 地名
    - Detail: 此作者在此地点发生的事件。以长字符串存储，包含诗词引用。可能发生的事件大于等于一条。实际处理时候需要去除引用的诗词，并使用Array存储多个事件。

由于数据处理的量级并不大，但是内容均十分复杂，人工数据处理是不可避免的事情***。***

***一种策略：通过人工处理一定数量的数据，先进行系统设计，等系统功能设计完成之后，更新数据到完整数据（这种数据从小到大的方法，一定程度上是可以说明系统的功能数据解耦合、工作流的自动化特性、可扩展性、OLAP的有效性）。***

## 原始轨迹数据展示与搜索。

在原始轨迹数据展示和搜索界面中，要求对于数据具备以下的特性:

### 数据层级关联性质

对于诗人而言，要求有诗人与朝代之间的对应关系：

* 即通过诗人可以检索到对应的朝代
* 可以通过朝代检索到所有诗人
* 每个朝代有一个default诗人供下钻使用

对于地点，需要有层级关系对应：

* 通过地点可以检索到对应的省份
* 通过省份可以检索到对应的国家
* 通过省份可以检索到所有其下的地点
* 每个省份有一个default地点供下钻使用
* 通过国家可以检索到其下所有的省份
* 每个国家有一个default省份供下钻使用

#### 补充说明

实践中，所有的default值全部取消。

### 按地点进行堆叠的数据

将诗人的人生轨迹数据按照地点进行重组堆叠，可以通过地点检索到所有的相关的诗人活动记录。

#### 补充说明

实践中，可以由计算引擎实时计算得到。

## 基于人生轨迹数据立方体的OLAM计算

### 数据选择部分

诗人、地点的映射关系可由上文完全覆盖。

实际传递到后端，可以考虑传递完整的诗人、地点List（考虑到浏览器URL长度限制，在此处可以使用body传参）。

时间关系是单年、十年、百年之间的映射关系。详细的计算方式为需要依据所有年份的起始和终止时间，其中十年和百年均以0为区分单位，不足部分可以舍弃处理（即十年一定是611~620、621~630、601~700这样的类型）。实际传参时可以按照年份范围进行传参。

### 图表可视化部分

所有图表的内容均需要按照数据选择部分进行后端的实时计算分析，且具体的数据内容以统计偏多，只能缓存，无法进行全部持久化存储。具体的数据设计部分可以参见后端的Swagger API文档。

## 动态柱形图可视化

此部分内容可以进行持久化，只需按照单个年份为单位，将各个内容的部分统计数据并持久化便可。

## 基于轨迹嵌入技术的时空流动性分析

按照对应的嵌入选取方式，计算整条轨迹的嵌入，轨迹点的嵌入。并将处理的分析结果持久化。

在数据库中存储时，仅需存储分析结果即可，无需存储嵌入数据。

## GL地球可视化

在此处需要将轨迹中的各个地点信息与时间进行绑定（**此处一定需要人工标注**）。

***注：实践中已替换GL地球为AMap Loca可视化。***

# 数据处理要求

## 项目Id编号说明

对每个数据子项目，均会进行数据Id编号处理。本节则分门别类对Id编号的规则进行说明。

* 诗人id：Integer ABC，ABC为按照爬取顺序增序设置的id编号，唐朝在前，宋朝在后
* 朝代id：Integer XX，XX为增序的朝代编号
* 国家id：String XXX，XXX为增序的国家编号
* 省份id：String XXXYYY，XXX为国家编号，YYY为国家下对应的增序省份编号
* 城市id：String XXXYYYZZZZ，XXX、YYY同上，ZZZZ为四位城市编号
* 地点id：String XXXYYYZZZZKKKK，XXX、YYY、ZZZZ同上，KKKK为四位的地点编号

所有的id可能需要多遍扫描来生成和填充（实际上只用了一遍就完成了），数据多遍处理并额外保存是很有必要的。

## 数据更新获取部分

无法从原数据中派生的数据，但可以借助自动化/半自动化手段获取的，均为需要更新获取的部分。

* 根据坐标的信息，反查所在的省份以及国家
  + 实际上由于包含很多在国外地点以及国内地点的化名，本流程可能仍然需要手工标注
* 坐标信息转换
  + 由于原数据的坐标为百度地图，而目的应用为高德地图，所以需要在所有坐标使用前进行坐标转换工作（可以使用高德地图api调用进行转换）。

## 数据人工标注部分

对于需要人工标注的部分，主要集中在诗人的人生轨迹对应上，每个地点的诗人访问过此处的年份拆分（需要手工标注）。此外需要人工标注的还有诗人对应的朝代信息。对每个诗人的简介以及在各个地点的简介静态文本内容拆分。

## 算法设计说明

***鉴于算法提供方的要求，只愿意提供处理完毕后的数据，故此部分不做说明与开源。***

# 基础数据格式

## 诗人数据格式草案

|  |
| --- |
| {    "id": 123123,    "name": "李白",    "dynasty": "唐",    "lifeDuration": [701, 762],    "mapConfig": {      "center": {        "lat": 42.8321571,        "lng": 75.28238      },      "scaleLevel": 5    },    "reference": [      "《李太白年谱补正》",      "《李白全集编年笺注》",      "《李太白全集校注》"    ],    "agesDetail": [      // one more objects will be included      {        "agesDuration": [1, 19],        "correspondPath": ["托克马克", "江油", "平武", "剑阁", "三台", "江油"]      }    ],    "travelDetail": {      "lines": [        // one more objects will be included        {          "markers": [            // one more objects will be included            {              "id": "ea80440f951041639f23f4b4fb38dbde",              "lat": 42.8321571,              "lng": 75.28238,              "regionId": "JP110",              "location": "托克马克",              "city": "",              "province": "楚河州",              "country": "吉尔吉斯斯坦",              "visitYear": 701            }          ]        }      ],      "markers": [        {          "lat": 42.8321571,          "lng": 75.28238,          "id": "",          "regionId": "JP110",          "location": "托克马克",          "city": "",          "province": "楚河州",          "country": "吉尔吉斯斯坦",          "visitTimes": 1,          "visitYears": [701],          "detail": [            // one more objects will be included            "701年 李白生于碎叶。随父居住。",            "702-704年　随父居。"          ]        }      ]    },    "trajectoryEmbedding": {      // define by the numbers of embedding kernel      "word2vec": {        "embedding": [],        "isNormal": true      },      "universalTransformer": {        "embedding": [],        "isNormal": true      },      "bert": {        "embedding": [],        "isNormal": true      }    }  } |

## 朝代数据格式草案

|  |
| --- |
| {    "唐": {      "id": 233,      "defaultPoet": "李白",      "poets": [        // for all poets        "杜甫",        "白居易"      ]    }    // for all dynasties  } |

## 地理数据格式草案

|  |
| --- |
| {    "中国": {      "id": 123,      "defaultProvince": "陕西",      "provincesList": [        // for all provinces      ],      "provinces": {        // one more keys will be included        "陕西": {          "id": 333,          "defaultCity": "",          "citiesList": [            // for all cities          ],          "cities": {            "西安": {              "id": 123332,              "defaultLocation": "咸阳",              "locationsList": [                // for all locations              ],              "locations": {                // one more keys will be included                "西安": {                  "id": 23334,                  "visitPoets": [                    // one more objects will be included                    {                      "name": "魏征",                      "visitYears": [618, 621, 622, 626, 638]                    }                  ],                  "locationEmbedding": {                    // define by the numbers of embedding kernel                    "word2vec": {                      "embedding": [],                      "poi": 100                    }                  }                }              }            }          }        }      }    }  } |

## 时间数据格式草案

|  |
| --- |
| {    "years": {      // all years      "618": {        "poets": [          // one more objects will be included          {            "label": "李白",            "number": 125          }        ],        "locations": [          // one more objects will be included          {            "label": "西安",            "number": 12          }        ],        "provinces": [          // one more objects will be included          {            "label": "陕西",            "number": 35          }        ],        "decade": [          // one more objects will be included          {            "label": "618-628",            "number": 76          }        ],        "century": [          // one more objects will be included          {            "label": "618-718",            "number": 76          }        ],        "frequentRoute": [          // one more objects will be included          {            "label": "西安-咸阳",            "number": 1          }        ]      }    }  } |

## 嵌入数据格式草案

|  |
| --- |
| {    "embedding": {      "word2vec": {        "poi": [],        "normal": [],        "abnormal": []      },      "universalTransformer": {        "poi": [],        "normal": [],        "abnormal": []      }    }  } |

# 数据库设计

数据库设计使用事实星座构型，数据库结构和数据参见数据处理代码仓库的dump文件。

# 补充说明

具体和详细的数据处理方法，可以参见数据处理代码仓库的各个README文件。